**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра Систем автоматизированного проектирования**

отчет

**по лабораторной работе №3**

**по дисциплине «Базы данных»**

Тема: Выполнение запросов по нескольким таблицам

|  |  |  |
| --- | --- | --- |
| Студенты гр. 2308 |  | Попов Н.А. |
|  |  | Бебия Р.А. |
|  |  | Чиков А.А. |
| Преподаватель |  | Горяинов С.В. |

Санкт-Петербург

2024

**Цель работы**

Научиться соединять данные из нескольких таблиц.

**Выполнение работы**

**Упражнение 1 – создание списка почтовой рассылки с использованием оператора JOIN.**

Требуется создать список рассылки читателей библиотеки. Список должен включать полное имя и информацию о месте жительства читателя

Запрос 1: Создание списка почтовой рассылки с использованием оператора JOIN.

SELECT firstname, ' ', middleinitial, ' ', lastname AS name, street, city, state, zip

FROM member

JOIN adult ON member.member\_no = adult.member\_no

Результат выполнения запроса представлен на Рисунке 1.

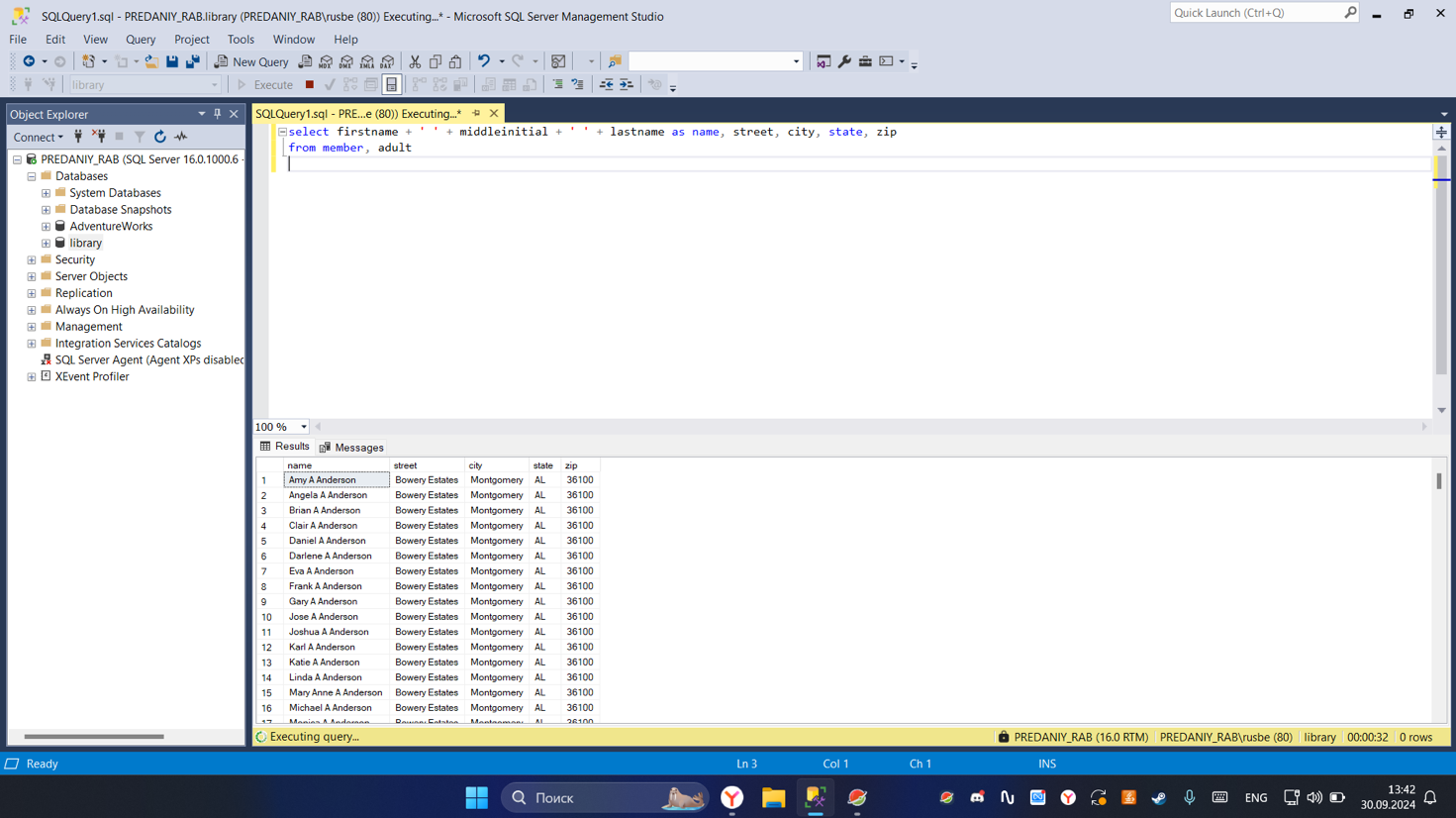


Рисунок 1 – Первые 10 строк результата выполнения запроса

**Упражнение 2 – объединение нескольких таблиц и сортировка результатов.**

Нужно выполнить запрос по таблицам title, item и copy, который возвращал бы поля isbn, copy\_no, on\_loan, title, translation и cover, а также строки из таблицы copy, где ISBN равен 1, 500 или 1000. Полученный набор должен быть отсортирован по полю isbn

Запрос 1: Объединение нескольких таблиц и сортировка результатов.

SELECT copy.isbn, copy\_no, on\_loan, title, translation, cover

FROM title, copy

INNER JOIN item on item.isbn = copy.isbn

WHERE copy.isbn = 1000 or copy.isbn = 1500

ORDER BY isbn;

Результат выполнения запроса представлен на Рисунке 2.

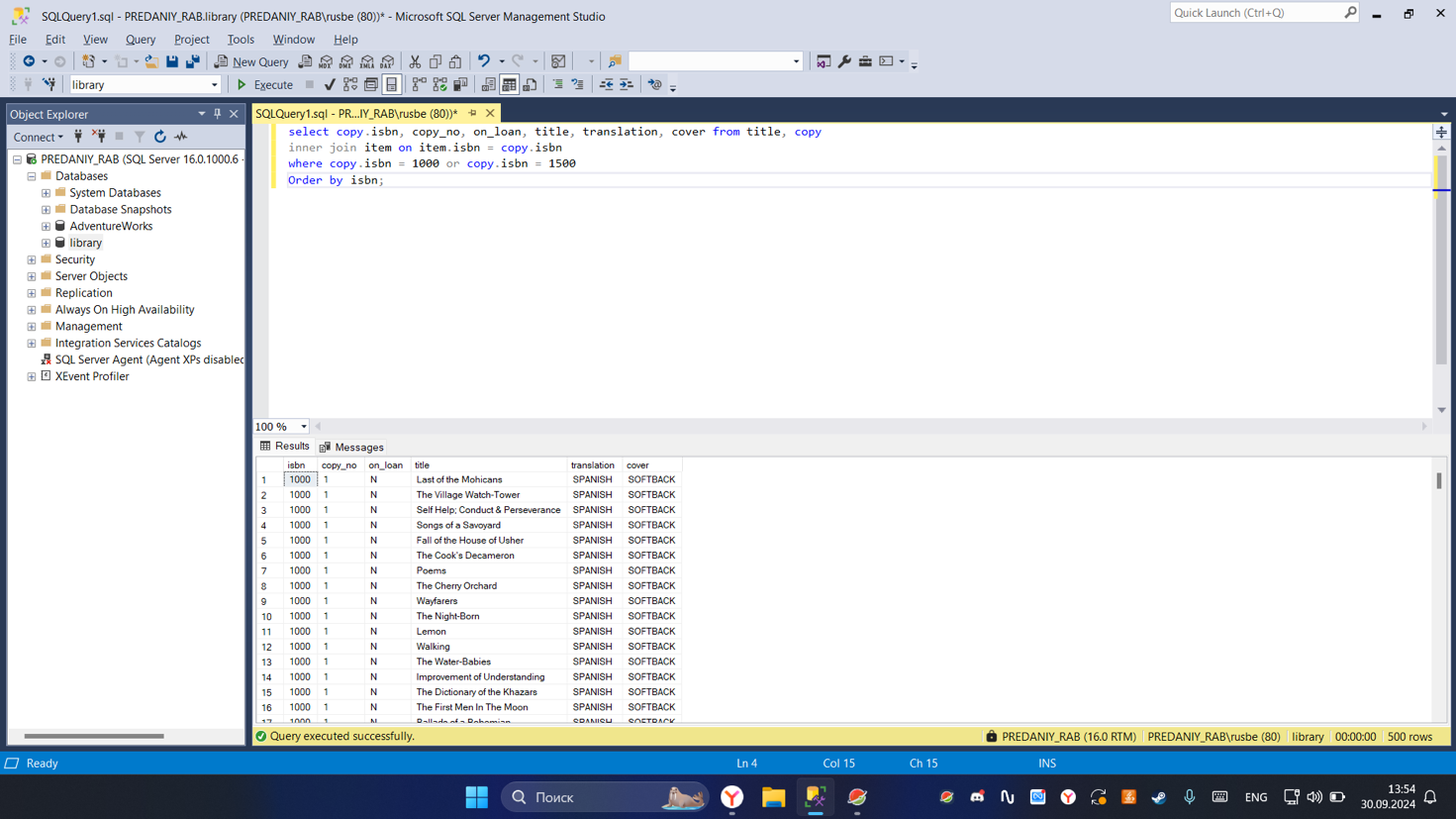


Рисунок 2 – Первые 10 строк результата выполнения запроса

**Упражнение 3 – объединение таблиц с использованием OUTER JOIN.** Необходимо выполнить запрос, возвращающий полное имя читателей

member\_no из таблицы member, isbn и log\_date из таблицы reservation для читателей с номерами 250, 341 и 1675. Результат отсортировать по member\_no. Показать информацию об этих читателях вне зависимости от того, взяты ими книги или нет.

Запрос 1: Объединение таблиц с использованием OUTER JOIN.

SELECT m.member\_no, CONCAT(m.lastname,' ', m.firstname,' ', m.middleinitial) AS name,

r.isbn, CONVERT(CHAR(8), r.log\_date, 112) AS date

FROM member AS m

LEFT OUTER JOIN reservation AS r ON m.member\_no = r.member\_no

WHERE m.member\_no IN (250, 341, 1675)

ORDER BY m.member\_no ASC;

Результат выполнения запроса представлен на Рисунке 3.

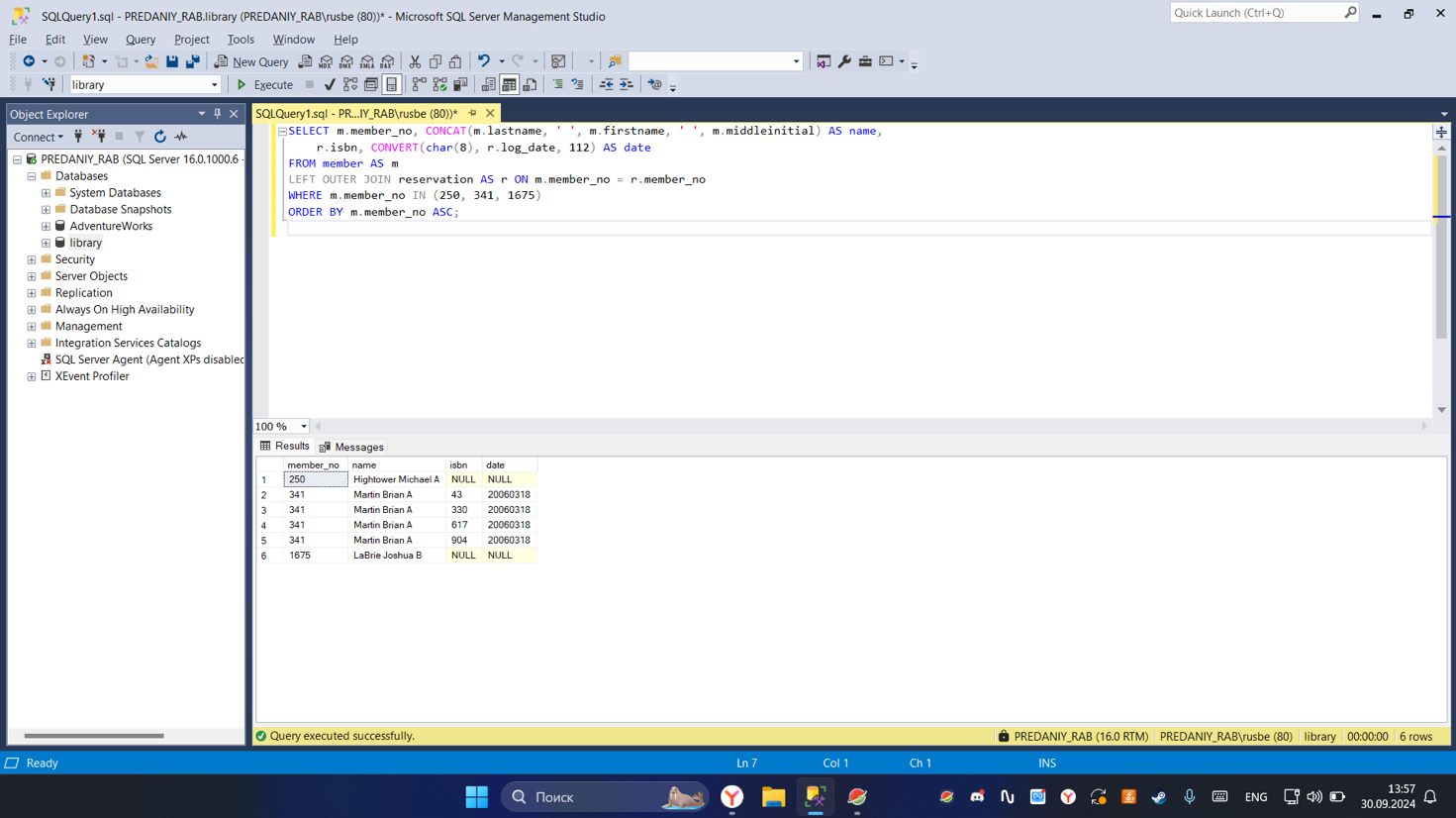


Рисунок 3 – Результат выполнения запроса

**Упражнение 4 – использование оператора UNION для соединения результирующих наборов**

Запрос 1: Определение читателей, живущих в Аризоне, у которых более 2 детей посещают библиотеку.

SELECT a.member\_no, COUNT(\*) AS numkids

FROM adult AS a

INNER JOIN juvenile AS j ON a.member\_no = j.adult\_member\_no

WHERE a.state = 'AZ'

GROUP BY a.member\_no

HAVING COUNT(\*) > 2

Результат выполнения запроса представлен на Рисунке 4.

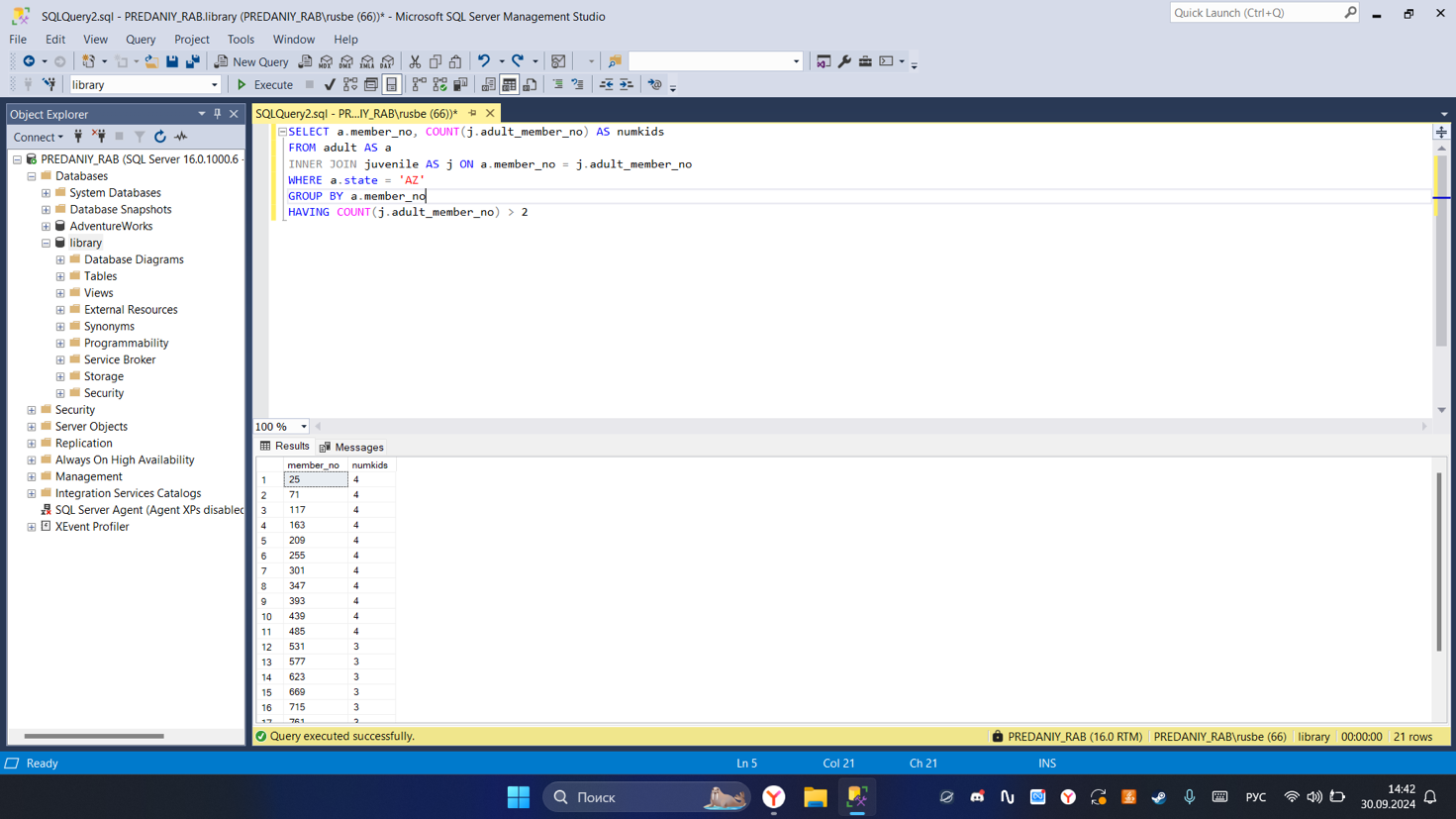


Рисунок 4 – Результат выполнения запроса

Запрос 2: Определение читателей, живущих в Калифорнии, у которых более 3 детей ходят в библиотеку.

SELECT a.member\_no, COUNT(j.adult\_member\_no) AS numkids

FROM adult AS a

INNER JOIN juvenile AS j ON a.member\_no = j.adult\_member\_no

WHERE a.state = 'CA'

GROUP BY a.member\_no

HAVING COUNT(j.adult\_member\_no) > 3

Результат выполнения запросов представлен на Рисунке 5.

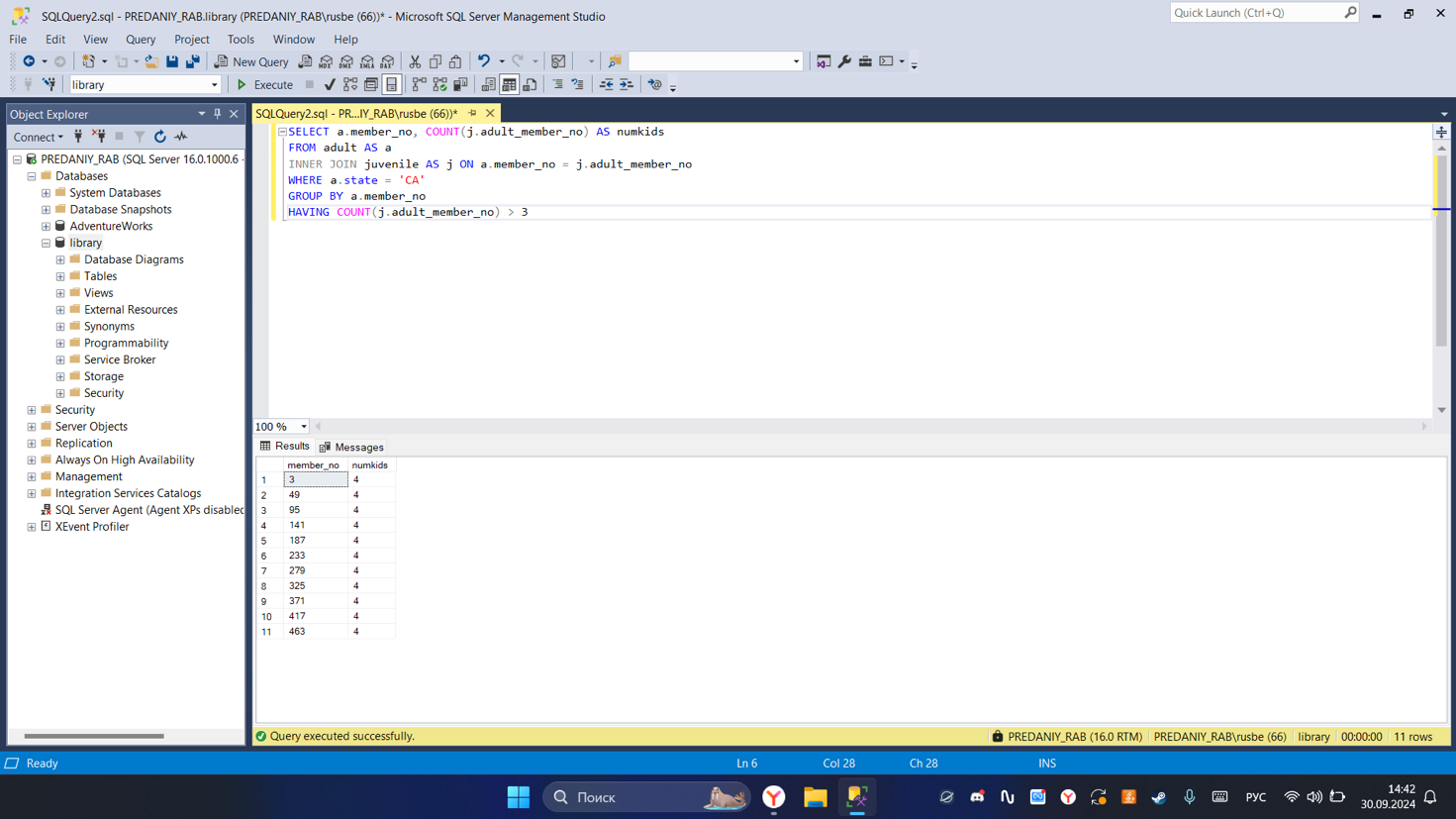


Рисунок 5 – Результат выполнения запроса

Запрос 3: Объединение результатов разных запросов.

SELECT a.member\_no, COUNT(j.adult\_member\_no) AS numkids

FROM adult AS a

INNER JOIN juvenile AS j ON a.member\_no = j.adult\_member\_no

WHERE a.state = 'AZ'

GROUP BY a.member\_no

HAVING COUNT(j.adult\_member\_no) > 2

UNION

SELECT a.member\_no, COUNT(j.adult\_member\_no) AS numkids

FROM adult AS a

INNER JOIN juvenile AS j ON a.member\_no = j.adult\_member\_no

WHERE a.state = 'CA'

GROUP BY a.member\_no

HAVING COUNT(j.adult\_member\_no) > 3;

Результат выполнения запроса представлен на Рисунке 6.

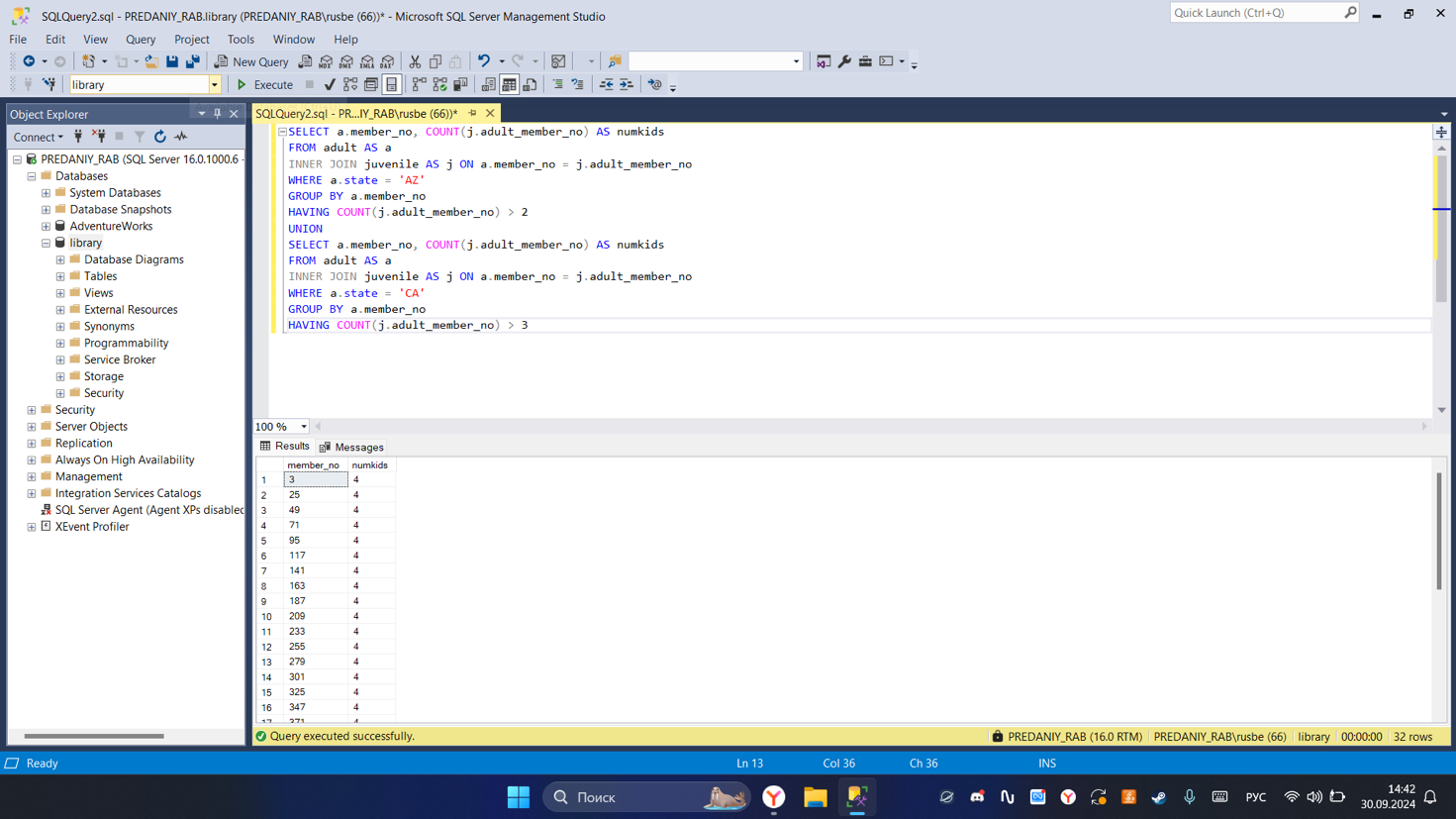


Рисунок 6 – Результат выполнения запроса

**Выводы**

В ходе лабораторной работы были изучены различные операторы объединения данных из нескольких таблиц, такие как JOIN, INNER JOIN, OUTER JOIN и UNION. Эти операторы позволяют гибко связывать и анализировать данные, находящиеся в разных таблицах базы данных.

Было рассмотрено применение INNER JOIN для получения пересечения данных, где совпадают ключевые значения в объединяемых таблицах. Это позволило создать выборки, отражающие связи между читателями и взятыми ими книгами. С помощью OUTER JOIN (левого, правого и полного) была достигнута возможность извлекать как совпадающие, так и отсутствующие данные, что обеспечило более полное представление о наличии книг у читателей, включая тех, у кого пока нет выданных книг.

Также использовался оператор UNION для объединения двух и более результирующих наборов в один, что позволило получить данные из различных запросов в едином формате. Это является незаменимым инструментом при создании комплексных отчетов, например, по активности читателей и статистике выдачи книг.

В результате лабораторной работы были приобретены навыки, которые позволяют эффективно управлять базами данных, работать с несколькими таблицами, извлекать и анализировать данные, формировать более информативные и разнообразные отчеты, что значительно облегчает процесс принятия решений на основе полученной информации.